Logistic Regression (2 class)

Bishop section 4.3

• Hypothesis is a “soft” hyperplane: of form \( w \cdot x \) (use “add a dimension” trick)
• Assume \( p(t = 1|x) \) is some \( f(w \cdot x) \), know \( f() \), learn \( w \)
• Experiment (to get data):
  – At the start, pick \( w \) from some prior
  – For each example:
    • pick \( x \)’s from some un-modeled \( P(x) \)
    • pick \( y = 1 \) with probability \( f(w \cdot x) \) (\( y = 0 \) otherwise)

• Learning Goal: learn \( w \)
• Model: discriminative not generative:
  – Model \( t \) as a function of \( x \), but not how \( x \)’s picked
Logistic Regression

• What should \( f(w \cdot x) \) be?
• Want confusion at \( w \cdot x = 0 \), more certainty away from boundary
• One \( f(w \cdot x) = \frac{\exp(w \cdot x)}{1 + \exp(w \cdot x)} \)
  \[ = \frac{1}{1 + \exp(-w \cdot x)} \]
Logistic Regression (2)

- Logistic regression finds maximum likelihood estimator - find \( w \) maximizing likelihood of \( w \) (given sample)
- Discriminative model:
  likelihood of \( w = p(\text{labels} \mid w, X) \)
- Find \( w \) maximizing \( p(\text{labels} \mid w, X) \)
Logistic regression 3

• Assume each \((x_i, t_i)\) drawn iid from some fixed but unknown distribution

• Maximize \(p(\text{labels} \mid w, \mathcal{X}) = \prod_i p(t_i \mid x_i, w)\)

• Equivalent to maximizing log-likelihood: 
  \[\sum_i \log( p(t_i \mid x_i, w) )\]
• Note:
  \[ p(y=1 \mid \mathbf{w}, \mathbf{x}) = f(\mathbf{w} \cdot \mathbf{x}) = \frac{1}{1+e^{-\mathbf{w} \cdot \mathbf{x}}} \]

  \[ p(y=0 \mid \mathbf{w}, \mathbf{x}) = 1 - f(\mathbf{w} \cdot \mathbf{x}) = \frac{e^{-\mathbf{w} \cdot \mathbf{x}}}{1+e^{-\mathbf{w} \cdot \mathbf{x}}} \]

• And:
  \[ p(y \mid \mathbf{w}, \mathbf{x}) = f(\mathbf{w} \cdot \mathbf{x})^y (1 - f(\mathbf{w} \cdot \mathbf{x}))^{(1-y)} \]

• Also, the derivative of sigmoid \( f(a) \) is:
  \[ f'(a) = f(a)(1-f(a)) \]
Logistic Regression 4

• Therefore, find $w$ maximizing

\[
\prod_t p(t_i | x_i, w)
\]

• Which is the the $w$ maximizing

\[
J(w) = \sum_i \log(p(t_i | x_i, w))
\]

• Take derivatives (some algebra)

\[
\frac{\partial J(w)}{\partial w_j} = \sum_i (y_i - p(t=1 | x_i, w)) x_{i,j}
\]

prediction error
Batch Gradient Ascent Alg

1. Initially $\mathbf{w}$ is all 0's
2. Compute gradient vector $\mathbf{g}$,
   For each $(x_i, t_i)$ example
   \[
   p_i = \frac{1}{1 + \exp(-\mathbf{w} \cdot \mathbf{x})} \quad \text{(predicted } p(t_i = 1), \text{ initially } 1/2) \]
   \[
   \text{error}_i = t_i - p_i
   \]
   for each feature $j$
   \[
   g_j = g_j + \text{error}_i \cdot x_{i,j}
   \]
3. Update $\mathbf{w} := \mathbf{w} + \eta \mathbf{g}$ \hspace{1cm} (\(\eta\) is step size)
4. Go to 2
Newton-Raphson (2\textsuperscript{nd} order)

- Want to find max of \( F(x) \)
  - Start with guess \( x_0 \)
  - Maximize \textit{second order approximation}
  - Iterate

\[
F(x + \delta) \approx F(x_0) + \delta F'(x_0) + \delta^2 F''(x_0)
\]

Max at: \( \delta = -F'(x_0) / F''(x_0) \)
For logistic regression:

- Each iteration of $2^{nd}$ Newton-Raphson is like a weighted least squares problem where weights depend on current ``guess'' for $w$
- Thus it is called *iteratively reweighted least squares*
• Learn weights $w_k$ for each class $k \in \{1, 2, ..., K\}$
• Class-$k$-ness of instance $x$ is $w_k^T x$
• Estimate $p(\text{Class} = k \mid x)$ for instance $x$ with SoftMax function:
\[
y_k(x) = \frac{\exp(w_k^T x)}{\sum_{j=1}^{K} \exp(w_j^T x)}
\]
• Want weights that maximize likelihood of the sample.
• Use one-of-\( K \) encoding for targets: each label \( t \) is a \( K \)-vector. All targets in sample is a \( N \times K \) matrix \( T \) where entry \( t_{n,k} \) is 1 iff \( k \) is the class of example \( n \).

• Likelihood of the sample is:

\[
p(T \mid w_1, w_2, \ldots, w_K) = \prod_{n=1}^{N} \prod_{k=1}^{K} \left( \sum_{t_{n,k} \in \{0,1\}} \left( \frac{1}{y_k(x_n)^{t_{n,k}}} \right) \right)
\]

• and negative log likelihood (cross entropy “error”) is

\[
- \ln (p(T \mid w_1, w_2, \ldots, w_K)) = - \sum_{n=1}^{N} \sum_{k=1}^{K} t_{n,k} \ln y_k(x_n)
\]

• Can be minimized with Newton-Raphson
Logistic Regression Summary

• Logistic regression gives distribution on labels: $p(y=1| \mathbf{x}, \mathbf{w})$
• Use gradient descent to learn $\mathbf{w}$
• $\mathbf{w} \cdot \mathbf{x}$ is equal to log odds: (exercise) $\log( p(y=1|\mathbf{w},\mathbf{x}) / p(y=0|\mathbf{w},\mathbf{x}) )$
• Can threshold at $\mathbf{w} \cdot \mathbf{x} = 0$ to get predictions
• With asymmetric loss can use different thresholds
Questions:

• What are strengths / weaknesses of LDA, Naïve Bayes, logistic regression?
• When might one perform better than another?
• How can you test which learning algorithm is better?
Exercises

• Run logistic regression in Weka on iris 2 data
• Compare Naïve Bayes and logistic regression results
## Comparison

<table>
<thead>
<tr>
<th></th>
<th>Fisher LDA</th>
<th>Perceptron</th>
<th>Logistic regression</th>
<th>Naïve Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Models</td>
<td>P(x</td>
<td>y)</td>
<td>?</td>
<td>P(y</td>
</tr>
<tr>
<td>Data</td>
<td>numeric</td>
<td>Numeric</td>
<td>Numeric</td>
<td>mixed</td>
</tr>
<tr>
<td>interpretable</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Somewhat</td>
</tr>
<tr>
<td>Missing values</td>
<td>yes(*)</td>
<td>No</td>
<td>No</td>
<td>yes</td>
</tr>
<tr>
<td>Outliers</td>
<td>Bad</td>
<td>Fatal(?)</td>
<td>good</td>
<td>Fair/poor</td>
</tr>
</tbody>
</table>

* (*) indicates optional or conditional support.
## Robustness

<table>
<thead>
<tr>
<th></th>
<th>LDA</th>
<th>Perceptron</th>
<th>Logistic regression</th>
<th>Naïve Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monotone transform</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>maybe</td>
</tr>
<tr>
<td>irrelevant features</td>
<td>Bad</td>
<td>Bad</td>
<td>Bad</td>
<td>some</td>
</tr>
<tr>
<td>Compute time</td>
<td>good</td>
<td>good</td>
<td>good(-)</td>
<td>v. good</td>
</tr>
</tbody>
</table>
Exercises (using iris2.arff)

• Duplicate an attribute 10 times, how does it affect algorithms?
• Add 10 random features (say 0,1), how does it affect algorithms?
• Cube an important feature, how does it affect hypothesis?